

データ転送制御装置、情報記憶媒体及び電子機器

発明の背景

発明の分野

5 本発明は、データ転送制御装置、情報記憶媒体及び電子機器に関し、特に、
バスに接続される複数のノード間で IEEE 1394などの規格に準じたデータ転送を行うためのデータ転送制御装置、情報記憶媒体及び電子機器に関する。

関連技術の説明

10 近年、IEEE 1394と呼ばれるインターフェース規格が脚光を浴びている。この IEEE 1394は、次世代のマルチメディアにも対応可能な高速シリアルバスインターフェースを規格化したものである。この IEEE 1394によれば、動画像などのリアルタイム性が要求されるデータも扱うことができる。また、IEEE 1394のバスには、プリンタ、スキャナ、CD-RWドライブ、ハードディスクドライブなどのコンピュータの周辺機器のみならず、ビデオカメラ、VTR、TVなどの家庭用電化製品も接続できる。このため、電子機器のデジタル化を飛躍的に促進できるものとして期待されている。

15 さて、この IEEE 1394においては、バスに電子機器が新たに接続されたり、バスから電子機器が取り外されたりして、バスに接続されるノードが増減すると、いわゆるバスリセットが発生する。そしてバスリセットが発生するとノードのトポロジ情報がクリアされ、その後、トポロジ情報が自動的に再設定される。即ち、バスリセットの発生後、ツリー識別（ルートノードの決定）、自己識別が行われ、その後、アイソクロナスリソースマネージャ等の管理ノードが決定される。そして通常のパケット転送が再開される。

20 25 このように IEEE 1394では、バスリセット後にトポロジ情報が自動的に再設定されるため、いわゆるホット状態でのケーブルの抜き差し（ホットプラグ）が可能となる。このため、一般ユーザは、VTRなどの通常の家庭用電化製品と同じように、電子機器へのケーブルの抜き差しを自由にできるようにな

り、いわゆるホームネットワークシステムの普及に役立つことができる。

しかしながら、IEEE 1394のバスに接続されたプリンタやスキャナなどのデバイスにおいて、このバスリセットの発生が要因となって以下のような問題が生じることが判明した。

5 即ち、IEEE 1394のバス上で印刷データの転送中にバスリセットが発生すると、パーソナルコンピュータなどのイニシエータは、印刷データの転送を最初から再度やり直す。従って、ターゲットであるプリンタに対して、印刷データの一部分だけが二重に送られてしまい、二重印刷などの誤印刷の問題が生じる。

10 また、スキャナでは、一旦ヘッドが動き出すと、ヘッドを元に戻して再度同じデータを取得することはできない。従って、バスリセットの発生後、イニシエータがデータ転送を最初から再度やり直そうとしても、データ転送を継続することができないという問題がある。

15 なお、バスリセットの発生により生じる不具合を解決する従来技術として、例えば特開平11-194902号公報に開示されるものがある。この従来技術では、バスリセットが発生すると、データ処理をホールドし、ネットワーク構成が再構築された後にデータ処理を再開する。

20 しかしながら、この従来技術では、バスリセット発生後に転送データを単に再送するだけであり、再送される転送データがバスリセット発生前の転送データの続きか否かの判断については行っていない。従って、この従来技術によっては二重印刷の問題を解決できない。

発明の要約

本発明は、以上のような技術的課題に鑑みてなされたものであり、その目的とするところは、ノードのトポロジ情報をクリアするリセットが発生した場合に生じる不具合を解消できるデータ転送制御装置、情報記憶媒体及び電子機器を提供することにある。

上記課題を解決するために本発明は、バスに接続される複数のノード間での

データ転送のためのデータ転送制御装置であって、相手ノードとの間でデータ転送が開始してから完了するまでの間のデータ転送期間中に、ノードのトポロジ情報をクリアするリセットが発生したか否かを判断する判断手段と、該リセットの発生前に相手ノードから転送されてきたデータ転送オペレーション要求のための第1のコマンドパケットの内容と、該リセットの発生後に相手ノードから転送されてきたデータ転送オペレーション要求のための第2のコマンドパケットの内容とを比較するコマンド比較手段と、ノードのトポロジ情報をクリアするリセットが前記データ転送期間中に発生したと判断され、且つ、前記第1、第2のコマンドパケットが同一内容であると判断された場合には、リセット発生時点のデータ転送の続きをデータ転送を再開する再開手段とを含むことを特徴とする。

本発明によれば、ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生したか否かが判断される。また、リセット発生前に転送されてきた第1のコマンドパケットの内容と、リセット発生後に転送されてきた第2のコマンドパケットの内容が比較される。そして、データ転送期間中にリセットが発生したと判断され、且つ、第1、第2のコマンドパケットが同一内容であると判断されると、リセット発生時点の続きを（例えばリセット発生時点で転送を完了したデータの次のデータから）、データ転送が再開されるようになる。

一方、例えば、第1、第2のコマンドパケットが同一内容でないと判断された場合等には、リセット発生後の第2のコマンドパケットは最初から処理されるようになる。

従って本発明によれば、相手ノードが、リセット発生後にリセット発生前と同一内容のコマンドパケットを転送要求してきた場合には、リセット発生時点の続きをデータ転送を再開できるようになる。従って、例えばデータ転送制御装置の上層のデバイスにデータが重複して転送されてしまい、上層のデバイスが誤動作するなどの問題を解消できる。

なお、前記判断手段は、ノードのトポロジ情報をクリアするリセットの発生

時点においてデータ転送オペレーション要求のための前記第1のコマンドパケットを処理中であり、且つ、該第1のコマンドパケットによるデータ転送が既に行われておりデータ転送完了のステータスを相手ノードに転送していない場合に、該リセットが前記データ転送期間中に発生したと判断することが望ましい。

5 また本発明は、前記判断手段が、ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生したと判断した場合に、データ転送が継続して再開される可能性があることを示す継続フラグをオンにセットすることを特徴とする。このようにすれば、継続フラグがオフの場合には、コマンドパケットの比較処理が行われないようにすることができる。従って、無用なコマンドパケットの内容比較処理が行われないようになり、処理負担の軽減化を図れる。

10 また本発明は、データ転送を再開するアドレスを特定するための情報と、データ転送オペレーション要求のための前記第1のコマンドパケットの内容とを、該リセットの発生後、データ転送が再開されるまでの間に記憶するコマンド記憶手段を含むことを特徴とする。このような情報を記憶しておけば、データ転送の再開処理を簡素な処理で実現できるようになる。

15 また本発明は、前記コマンド比較手段が、ノードのトポロジ情報をクリアするリセットの発生後に相手ノードから転送されてきたコマンドパケットの中で最初に転送されてきたデータ転送オペレーション要求のためのコマンドパケットを、前記第1のコマンドパケットの比較対象となる前記第2のコマンドパケットとして採用することを特徴とする。このようにすれば、最初のデータ転送オペレーション要求のためのコマンドパケットが転送されてくるまで、コマンドパケットの内容比較処理が実行されるのが、繰り越されるようになる。これにより、無用なコマンドパケットの内容比較処理が行われるのを防止でき、処理負担の軽減化を図れる。

20 25 また本発明は、データ転送完了のステータスを相手ノードに転送したが、ノードのトポロジ情報をクリアするリセットの発生が要因となって相手ノードからアクノリッジメントが返って来なかつた場合に、データ転送不可状態に状態遷移することを特徴とする。このように、相手ノードからアクノリッジメント

が返って来なかつた場合には、相手ノードがステータスを受け取つたか否かが不明となる。従つて、このような場合にリセット発生時点の続きたからデータ転送を開始してしまうと、誤ったデータ転送が行われてしまう可能性がある。本発明によれば、ノードのトポロジ情報をクリアするリセットが要因となって相手ノードからアクノリッジメントが返つて来なかつた場合には、データ転送不可状態に状態遷移するため、このような誤ったデータ転送が行われる事態を防止できる。

また本発明は、上層のデバイスから転送されてきた転送データの中で、ノードのトポロジ情報をクリアするリセットの発生時点において相手ノードへ未だ転送していない転送データについては、破棄せずに保持しておくことを特徴とする。このようにすれば、スキヤナ等により取り込んだデータが、リセットの発生により失われてしまう等の不具合を防止できる。

なお本発明では、前記リセットが、IEEE 1394の規格において定義されるバスリセットであることが望ましい。

また本発明は、上記のいずれかのデータ転送制御装置との間でのデータ転送を制御するためのプログラムを含む情報記憶媒体であつて、ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した場合には、該リセットの発生前に転送されたデータ転送オペレーション要求のための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする。このようにすれば、誤ったデータ転送再開処理が行われてしまう事態を防止でき、リセットの発生が要因となる不具合の発生を防止できる。

また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して相手ノードから受信したデータに所与の処理を施す装置と、処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とする。また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して相手ノードに送信するデータに所与の処理を施す装置と、処理が施されるデータを取り込む

ための装置とを含むことを特徴とする。

本発明によれば、ノードのトポロジー情報をクリアするリセットの発生によりシステムに不具合が生じる事態を防止でき、電子機器が誤動作するのを防止できる。またデータ転送の高速化を図れ、電子機器の低コスト化、電子機器の処理の高速化なども図ることができる。

5

図面の簡単な説明

図1は、IEEE1394の層構造について示す図である。

図2は、SBP-2について説明するための図である。

10 図3は、SBP-2のデータ転送処理の概略について説明するための図である。

図4は、データをイニシエータからターゲットに転送する場合のコマンド処理について説明するための図である。

図5は、データをターゲットからイニシエータに転送する場合のコマンド処理について説明するための図である。

15 図6A、図6B、図6Cは、ページテーブルについて説明するための図である。

図7は、本実施形態のデータ転送制御装置の構成例を示す図である。

図8は、ターゲット側（ファームウェア）の処理の概要を示すフローチャートである。

20 図9は、イニシエータ側（デバイスドライバ）の処理の概要を示すフローチャートである。

図10A、図10B、図10Cは、二重印刷の問題について説明するための図である。

25 図11は、データ転送期間中にバスリセットが発生し、且つ、バスリセットの前後でORBの内容が同一の場合に、データ転送を継続して再開する手法について説明するための図である。

図12は、ORBの内容比較について説明するための図である。

図13は、継続フラグについて説明するための図である。

図14は、データ転送を再開するアドレスを特定するための情報について説明するための図である。

図15は、印刷コマンドを含む最初のコマンドブロックORBが来るまで、コマンド比較処理を繰り越す手法について説明するための図である。

図16は、ステータスのライト中にバスリセットが発生し、ACKミッシングとなつた場合に、テッド状態に移行する手法について説明するための図である。

図17は、バスリセット時にイニシエータに未だ転送していないデータを破棄せずに保持しておく手法について説明するための図である。

図18は、バスリセット発生時(リコネクト時)の本実施形態の詳細な処理例について示すフローチャートである。

図19は、バスリセット発生時(リコネクト時)の本実施形態の詳細な処理例について示すフローチャートである。

図20は、バスリセット発生時(リコネクト時)の本実施形態の詳細な処理例について示すフローチャートである。

図21は、通常時の本実施形態の詳細な処理例について示すフローチャートである。

図22は、通常時の本実施形態の詳細な処理例について示すフローチャートである。

図23A、図23B、図23Cは、種々の電子機器の内部ブロック図の例である。

図24A、図24B、図24Cは、種々の電子機器の外観図の例である。

発明の好適な実施例の説明

以下、本発明の好適な実施形態について図面を用いて詳細に説明する。

1. IEEE 1394

まず、IEEE 1394について簡単に説明する。

1. 1 概要

IEEE 1394 (IEEE 1394-1995、P1394.a) では 100～400Mbps の高速なデータ転送が可能となっている (P1394.b では 800～3200Mbps)。また、転送速度が異なるノードをバスに接続することも許される。

5 各ノードはツリー状に接続されており、1つのバスに最大で 63 個のノードが接続可能になっている。なお、バスブリッジを利用すれば約 64000 個のノードを接続することも可能である。

10 IEEE 1394 では、パケットの転送方式として非同期転送とアイソクロナス転送が用意されている。ここで非同期転送は、信頼性が要求されるデータの転送に好適な転送方式であり、アイソクロナス転送は、リアルタイム性が要求される動画像や音声などのデータの転送に好適な転送方式である。

1. 2 層構造

IEEE 1394 の層構造 (プロトコル構成) を図 1 に示す。

15 IEEE 1394 のプロトコルは、トランザクション層、リンク層、物理層により構成される。また、シリアルバスマネージメントは、トランザクション層、リンク層、物理層をモニターしたり制御したりするものであり、ノードの制御やバスのリソース管理のための種々の機能を提供する。

20 トランザクション層は、上位層にトランザクション単位のインターフェース (サービス) を提供し、下層のリンク層が提供するインターフェースを通して、リードトランザクション、ライトトランザクション、ロックトランザクション等のトランザクションを実施する。

25 ここで、リードトランザクションでは、応答ノードから要求ノードにデータが転送される。一方、ライトトランザクションでは、要求ノードから応答ノードにデータが転送される。またロックトランザクションでは、要求ノードから応答ノードにデータが転送され、応答ノードがそのデータに処理を施して要求ノードに返信する。

リンク層は、アドレッシング、データチェック、パケット送受信のためのデータフレーミング、アイソクロナス転送のためのサイクル制御などを提供する。

物理層は、リンク層により使用されるロジカルシンボルの電気信号への変換や、バスの調停や、バスの物理的インターフェースを提供する。

1. 3 SBP-2

さて、図2に示すように、IEEE 1394のトランザクション層の一部の機能を含む上位のプロトコルとして、SBP-2 (Serial Bus Protocol-2) と呼ばれるプロトコルが提案されている。

ここでSBP-2は、SCSIのコマンドセットをIEEE 1394のプロトコル上で利用可能にするために提案されたものである。このSBP-2を用いれば、既存のSCSI規格の電子機器で使用されていたSCSIのコマンドセットに最小限の変更を加えて、IEEE 1394規格の電子機器に使用できるようになる。従って、電子機器の設計や開発を容易化できる。また、SCSIのコマンドだけではなく、デバイス固有のコマンドもカプセル化して利用できるため、非常に汎用性が高い。

図3に示すようにSBP-2では、まず、イニシエータ（例えばパーソナルコンピュータ）により作成されたログインORB (Operation Request Block) を用いてログイン処理が行われる（ステップT1）。次に、ダミーORBを用いてフェッчエージェントの初期化が行われる（ステップT2）。そして、コマンドブロックORB (ノーマルコマンドORB) を用いてコマンド処理が行われ（ステップT3）、最後に、ログアウトORBを用いてログアウト処理が行われる（ステップT4）。

ここで、ステップT3のコマンド処理においては、図4のA1に示すように、イニシエータがライト要求パケットを転送して（ライト要求トランザクションを発行して）、ターゲットのドアベルレジスタをリングする。すると、A2に示すように、ターゲットがリード要求パケットを転送し、イニシエータが対応するリード応答パケットを返す。これにより、イニシエータが作成したORB (コマンドブロックORB) が、ターゲットのデータバッファにフェッчされる。そして、ターゲットは、フェッчされたORBに含まれるコマンドを解析する。

そして、ORBに含まれるコマンドがSCSIのライトコマンドであった場合には、A3に示すように、ターゲットがリード要求パケットをイニシエータに転送し、イニシエータが対応するリード応答パケットを返す。これにより、イニシエータのデータバッファに格納されているデータがターゲットに転送される。そして、例えばターゲットがプリンタであった場合には、転送されたデータがプリンタエンジンにより印刷される。

一方、ORBに含まれるコマンドがSCSIのリードコマンドであった場合には、図5のB1に示すように、ターゲットは、一連のライト要求パケットをイニシエータに転送する。これにより、例えばターゲットがスキャナであった場合には、スキャナエンジンにより取得されたスキャンデータが、イニシエータのデータバッファに転送されることになる。

このSBP-2によれば、ターゲットは、自身が都合の良いときに要求パケットを転送して（トランザクションを発行して）、データを送受信できる。従って、イニシエータとターゲットが同期して動く必要がなくなるため、データ転送効率を高めることができる。

なお、IEEE1394の上位プロトコルとしては、SBP-2以外にも、FCP (Function Control Protocol) と呼ばれるプロトコルなども提案されている。

さて、ターゲット、イニシエータ間でデータ転送を行う場合、図6Aのよう20にイニシエータ（相手ノード）のデータバッファ（記憶手段）にページテーブルが存在する場合と、存在しない場合がある。

そして、ページテーブルが存在する場合には、図6Bに示すように、イニシエータが作成したORBの中には、そのページテーブルのアドレスやエレメント数が含まれる。そして、転送データのアドレス（読み出しアドレス、書き込みアドレス）は、このページテーブルを用いて間接アドレス指定される。

一方、ページテーブルが存在しない場合には、図6Cに示すように、ORBの中にアドレスとデータ長が含まれ、転送データのアドレスが直接アドレス指定される。

1. 4 バスリセット

IEEE 1394では、電源が投入されたり、途中でデバイスの抜き差しが発生すると、バスリセットが発生する。即ち、各ノードは、ポートの電圧変化を監視している。そして、バスに新たなノードが接続されるなどしてポートの電圧に変化が生じると、この変化を検知したノードは、バス上の他のノードに対して、バスリセットが発生したことを知らせる。また、各ノードの物理層は、バスリセットが発生したことをリンク層に伝える。

そして、このようにバスリセットが発生すると、ノードIDなどのトポロジ情報がクリアされる。そして、その後、トポロジー情報が自動的に再設定される。即ち、バスリセット後、ツリー識別、自己識別が行われる。その後、アイソクロナスリソースマネージャ、サイクルマスター、バスマネージャ等の管理ノードが決定される。そして、通常のパケット転送が再開される。

このようにIEEE 1394では、バスリセット後にトポロジ情報が自動的に再設定されるため、電子機器のケーブルを自由に抜き差しできるようになり、いわゆるホットプラグを実現できる。

なお、トランザクションの途中でバスリセットが発生した場合には、そのトランザクションはキャンセルされる。そして、キャンセルされたトランザクションを発行した要求ノードは、トポロジー情報が再設定された後に、要求パケットを再度転送する。また、応答ノードは、バスリセットによりキャンセルされたトランザクションの応答パケットを要求ノードに返送してはならない。

2. 全体構成

次に、本実施形態のデータ転送制御装置の全体構成例について図7を用いて説明する。なお、以下では、イニシエータとの間でデータ転送を行うターゲットがプリンタである場合について例にとり説明するが、本発明はこれに限定されない。

本実施形態のデータ転送制御装置10は、PHYデバイス12（物理層のデバイス）、リンクデバイス14（リンク層のデバイス）、CPU16（プロセッサ）、データバッファ18（記憶手段）、ファームウェア20（処理手段）

を含む。なお、PHYデバイス12、リンクデバイス14、CPU16、データバッファ18は、任意の構成要素であり、本実施形態のデータ転送制御装置10は、これらの構成要素を全て含む必要はない。

PHYデバイス12は、図1の物理層のプロトコルをハードウェアにより実現するための回路であり、リンクデバイス14により使用されるロジカルシンボルを電気信号に変換する機能を有する。

リンクデバイス14は、図1のリンク層のプロトコルやトランザクション層のプロトコルの一部をハードウェアにより実現するための回路であり、ノード間でのパケット転送のための各種サービスを提供する。

CPU16は、装置全体の制御やデータ転送の制御を行うものである。

データバッファ18は、転送データ（パケット）を一時的に格納するバッファであり、SRAM、SDRAM、或いはDRAMなどのハードウェアにより構成される。なお、本実施形態では、データバッファ18は、ランダムアクセス可能なパケット記憶手段として機能する。

ファームウェア20は、CPU16上で動作する種々の処理ルーチン（処理モジュール）を含むプログラムであり、トランザクション層のプロトコルは、このファームウェア20と、ハードウェアであるCPU16等により実現される。

なお、イニシエータであるパーソナルコンピュータ100が含むデバイスドライバ102は、周辺機器を管理制御するための種々の処理ルーチンを含むプログラムである。このプログラムは、情報記憶媒体110（FD、CD-ROM、DVD、ROM）を利用してパーソナルコンピュータ100にインストールされる。

ここで、デバイスドライバ102のプログラムは、ホストシステムが有する情報記憶媒体（ハードディスク、磁気テープ等）からインターネットなどのネットワークを介してダウンロードし、パーソナルコンピュータ100にインストールするようにしてもよい。このようなホストシステムが有する情報記憶媒体の使用も本発明の範囲内に含まれる。

ファームウェア20(F/W)は、コミュニケーション部30(COM)、マネージメント部40(MNG)、プリントタスク部50(PRT)、フェッチ部60(FCH)を含む。

ここで、コミュニケーション部30は、リンクデバイス14などのハードウェアとの間のインターフェースとして機能する処理モジュールである。

マネージメント部40(マネージメントエージェント)は、ログイン、リコネクト、ログアウト、リセット等の管理を行う処理モジュールである。例えばイニシエータがターゲットにログインを要求した場合には、まず、このマネージメント部40が、このログイン要求を受け付けることになる。

プリントタスク部50は、後段のアプリケーション層(上層)であるプリントエンジンとの間のデータ転送処理を行う処理モジュールである。

フェッチ部60(フェッチエージェント、コマンドブロックエージェント)は、コマンドブロックORBが含むコマンドを実行するための処理モジュールである。フェッチ部60は、単一の要求しか扱うことができないマネージメント部40と異なり、イニシエータからの要求により自身がフェッチしたORBのリンクリストも扱うことができる。

フェッチ部60は、判断部62、コマンド記憶部64、コマンド比較部66、アドレス記憶部68、アドレス比較部70、データ転送再開部72を含む。

ここで判断部62は、イニシエータ(相手ノード)との間で印刷データを転送するデータ転送期間中に、バスリセット(広義には、ノードのトポロジ情報をクリアするリセット)が発生したか否かを判断する処理を行う。

コマンド記憶部64は、バスリセットの発生前にイニシエータから転送されたORB(コマンドブロックORB。広義には、データ転送オペレーション要求のためのコマンドパケット)の内容を、バスリセットが発生した時点やリコネクトが成功した時点などで記憶するための処理を行う。

コマンド比較部66は、バスリセットの発生前にイニシエータから転送されたORB(コマンドブロックORB)の内容(コマンド記憶部64により記憶された内容)と、バスリセットの発生後にイニシエータから転送されてき

たORBの内容とを比較する処理を行う。

アドレス記憶部68は、イニシエータとの間で転送される転送データ（印刷データ）の先頭アドレス（第1のアドレス）を記憶するための処理を行う。

アドレス比較部70は、バスリセットが発生した場合に、アドレス記憶部68により記憶された先頭アドレス（第1のアドレス）と、バスリセット発生後の転送データの先頭アドレス（第2のアドレス）とを比較する処理を行う。

データ転送再開部72は、バスリセットがデータ転送期間中に発生したと判断され、且つ、ORB（コマンドブロックORB）の内容が一致したと判断された場合に、バスリセット発生時点のデータ転送の続き（バスリセット発生時点で転送したデータの次のデータ）からデータ転送を再開する処理を行う。

3. 処理の概要

次に、本実施形態の処理の概要について説明する。

図8は、ターゲット側（ファームウェア）の処理の概要について示すフローチャートである。

イニシエータから印刷要求があると、ターゲットは、イニシエータのデータバッファからORBをリードする（ステップS1）。そして、ページテーブルが存在する場合には、ORBに含まれるページテーブルアドレス（図6B参照）に基づいて、イニシエータのデータバッファからページテーブルをリードする（ステップS2）。次に、リードしたページテーブルに基づいてイニシエータのデータバッファから印刷データをリードする（ステップS3）。そして、ページテーブルにより指定される印刷データを全てリードすると、ステータスをライトして、データ転送が成功したか否かなどのステータスをイニシエータに伝える（ステップS4）。以上の処理を、全ての印刷データが転送されるまで繰り返す（ステップS5）。

そして本実施形態では、印刷データの転送中（データ転送期間）にバスリセットが発生すると、リコネクト後の最初の印刷要求時に以下の処理を行う。

即ち、まず、バスリセット前のORBの内容や印刷データの先頭アドレスと、バスリセット後のORBの内容や印刷データの先頭アドレスとが同一か否かを

判断する（ステップS 6）。そして、同一であると判断した場合には、バスリセット発生時点の続きからデータ転送を再開する（ステップS 7）。一方、同一でないと判断した場合には、バスリセット後のORBを新規のORBとして最初から処理する（ステップS 8）。

5 図9は、イニシエータ側（デバイスドライバ）の処理の概要について示すフローチャートである。

10 アプリケーションプログラムからの印刷ジョブが発生すると、イニシエータは、印刷のためのORBやページテーブルを作成し、データバッファに書き込む（ステップS 10）。次に、作成したORBをリードするようにターゲットに対して指示する（ステップS 11。図4のA1参照）。

15 次に、バスリセットが発生したか否かを判断し（ステップS 12）、発生しなかった場合には、ステータスがターゲットから送られてきたか否かを判断する（ステップS 13）。そして、送られてきた場合には、全ての印刷データが転送されたか否かを判断し（ステップS 14）、転送されていない場合には、ステップS 10に戻り、転送された場合には印刷ジョブを終了する。

20 そして本実施形態では、ステップS 12でバスリセットが発生したと判断されると、イニシエータが、ORB、ページテーブルを再作成し（ステップS 15）、再作成したORBをリードするようにターゲットに対して指示する（ステップS 11）。この場合にイニシエータ（デバイスドライバ）は、バスリセット発生前のORBの内容や印刷データの先頭アドレスと、バスリセット発生後のORBの内容や印刷データの先頭アドレスとが同一になるように、ORBを再作成する。

4. 本実施形態の特徴

25 さて、印刷データの転送中にバスリセットが発生すると、以下のような問題が生じることが判明した。

例えば図10Aに示すように、C1に示す位置（アドレス）までデータを転送したところで、バスリセットが発生したとする。この場合には、バスリセット発生時点で処理中であったトランザクションは全てキャンセルされる。従つ

て、バスリセット前に印刷データの転送を要求していたイニシエータは、図 1
10 B に示すように、バスリセット後に印刷のための ORB を再度作成して、印
刷データの転送を最初からやり直すようにターゲットに指示する。このため、
図 10 B の C 2 に示す位置からデータ転送が再開されてしまい、印刷データの
一部だけが二重に送られてしまう。この結果、図 10 C に示すような二重印
5 刷の問題が発生する。

このような問題を解決するために、本実施形態では、以下に説明するような
手法を採用している。

即ち本実施形態では、まず、図 11 の D 1 に示すように、データ転送期間中
10 (イニシエータとの間で印刷データの転送が開始してから完了するまでの期
間) にバスリセットが発生したか否かを判断する。より具体的には、バスリセ
ットの発生時点において、D 2 に示す ORB (コマンドブロック ORB) を処
理中であり、且つ、その ORB によるデータ転送が既に行われており、D 3 に
示すデータ転送完了のステータスをイニシエータに転送していない場合に、バ
スリセットがデータ転送期間中に発生したと判断する。

また本実施形態では、図 11 の D 4 に示すように、バスリセットの発生後に
イニシエータがリコネクトに成功し、新たな ORB を作成して転送要求してき
た場合に、バスリセット前の ORB (第 1 のコマンドパケット) の内容とバス
リセット後の ORB (第 2 のコマンドパケット) の内容を比較する。

20 そして、データ転送期間中にバスリセットが発生したと判断され、且つ、バ
スリセット前の ORB の内容とバスリセット後の ORB の内容が同一内容と判
断された場合には、D 5 に示すように、バスリセット発生時点のデータ転送の
続きからデータ転送を再開する (図 8 のステップ S 7 参照)。即ち、バスリセ
ット発生時点で既に転送を完了していたデータの次のデータからデータ転送を
25 再開する。

一方、データ転送期間中にバスリセットが発生しなかった場合や、バスリセ
ットの前と後で ORB の内容が同一でない場合には、バスリセット後の ORB
を新規のものとして最初から処理する (図 8 のステップ S 8 参照)。

このようにすることで、図11のD6に示す部分の転送データが、図10Bの場合と異なり、二重転送されないようになる。従って、図10Cに示すような誤印刷が生じなくなる。また二重転送を避けることができるため、転送時間も短縮できる。

5 例えば、データ転送期間以外の期間にバスリセットが発生した場合には、D5のようにバスリセット発生時点の続きからデータ転送を再開しなくても、転送データが二重転送されることはないと問題はない。また、データ転送期間以外の期間にバスリセットが発生した場合には、データ転送の再開処理を行うよりも、ORBを最初から処理した方が、処理も簡素になり、処理負担も軽くなる。

また、本実施形態と異なる手法として、ORBの内容の比較処理を行わずに、常に、バスリセット発生時点の続きからデータ転送を再開するという手法も考えられる。

15 しかしながら、この手法によると、例えば、バスリセット後にイニシエータが印刷データの転送処理をキャンセルし、バスリセット前と全く異なるORBを作成した場合にも、図11のD5からデータ転送が再開されてしまうという不具合が生じる。

これに対して本実施形態では、ORBの内容がバスリセットの前後で同一の場合には、図11のD5からデータ転送が再開するが、同一でない場合には、全く新規のORBとして処理されるため、上記のような不具合が生じない。

20 なお本実施形態では、ORBの内容比較の際に種々の情報を比較している。例えば図12に示すように、本実施形態では、コマンドブロックORBが含むページテーブル存在フラグPや、データサイズや、コマンドブロック（コマンドセット）フィールドの中のオペレーションコード（印刷コマンド、リードコードなどを区別するコード）やデータ長を比較している。またORBが、ORMANDを識別するための識別情報（例えば順序番号）を含む場合には、この識別ORBを識別するための識別情報（例えば順序番号）を含む場合には、この識別情報も比較している。このような情報を比較することで、バスリセット前後のORBが同一か否かを簡素な処理で確実に判断できるようになる。

さて本実施形態では、図13のE1に示すように、バスリセットがデータ転送期間中に発生したと判断した場合には、データ転送が継続して再開される可能性があることを示す継続フラグをオンにセットするようにしている。例えば、バスリセットの発生後、リコネクトが成功した時などに、バスリセットがデータ転送期間中に発生したか否かが判断され、発生したと判断された場合に継続フラグがオンにセットされる。

そして図13のE2に示すように、ORBの内容比較は、この継続ビットがオンであることを条件に行われる。即ち、継続フラグがオフの場合には、ORBの内容比較は行われない。これにより、無用なORBの内容比較処理が行われることが防止されるようになり、ファームウェアの処理負担を軽減化できる。

また本実施形態では、データ転送を再開するアドレス（図11のD5）を特定するための情報や、バスリセット前のORBの内容などを、バスリセット発生時点やリコネクトが成功した時点（広義には、バスリセットの発生後、データ転送が再開されるまでの間）に記憶している。このような情報を記憶しておくことで、バスリセット後にイニシエータからORBが再度転送されてきた時に、バスリセット発生時点の続きをからデータ転送を再開するか否かの判断を容易に行えるようになる。またデータ転送の再開処理も簡素化できるようになる。

なお、データ転送を再開するアドレスを特定するための情報としては、具体的には次のような情報を考えることができる。

例えば図14に示すように、バスリセット発生時点において、ページテーブルのN番目のセグメントSEG Nが指定する転送データを転送していたとする。この場合には、データ転送を再開するアドレス（F1に示すアドレス）を特定するための情報としては、ページテーブルの先頭セグメントSEG 1に格納されるアドレス（転送データの先頭アドレス）や、セグメントSEG 1～SEG (N-1)により転送されたデータのサイズDS 1や、セグメントNのセグメント番号や、バスリセット発生時点までにセグメントNにより転送されたデータのサイズDS 2などを考えることができる。

また本実施形態では、バスリセット発生後にイニシエータから転送されてき

たORBの中で、最初に転送されてきた印刷コマンドを含むORB（コマンドブロックORB）を、バスリセット前のORBの比較対象となるORBとして採用するようにしている。

例えば図15に示すように、バスリセット後に、印刷コマンドを含むコマンドブロックORBが直ぐに転送されてくるとは限らない。即ち、イニシエータが、印刷コマンドを含むコマンドブロックORBを転送要求する前に、ダミーORBやマネージメントORBを転送要求してくる場合がある。或いは、プリンタのステータス等を調べるためにリードコマンドを含むコマンドブロックORBを転送要求してくる場合もある。

本実施形態では図15のG1に示すように、ダミーORB等が転送されてきても、ORBの内容比較処理を行わず、印刷コマンドを含むコマンドブロックORBが転送されてきた時に初めてORBの内容比較処理を行う。即ち、印刷コマンドを含む最初のコマンドブロックORBが来るまで、ORBの内容比較処理の実行を繰り越す（順送りにする）。これにより、無用なORBの内容比較処理が行われるのが防止され、ファームウェアの処理負担を軽減化できる。

さて、バスリセットの発生時期は全くの任意である。従って、例えば図16に示すように、ターゲットがデータ転送完了のステータスをイニシエータに転送したが、バスリセットの発生が要因となってイニシエーターからACK（ACKコンプリート）が返って来ず、ACKミッシングになる場合がある。

このような場合には、バスリセットの発生が要因となってイニシエータがステータスを受け取れず、ACKミッシングになったという第1のケースと、イニシエータはステータスを受け取り、ACKを返したが、バスリセットの発生が要因となってACKミッシングになったという第2のケースが考えられる。

そして、上記第1のケースでは、イニシエータはデータ転送が不成功であつたと考え、バスリセット後に同一のORBを再度作成するという第1の処理を行う。一方、上記第2のケースでは、イニシエータはデータ転送が成功したと考え、バスリセット後に次のORBを作成するという第2の処理を行う。

ところが、ターゲットにはACKミッシングであったという情報しか伝わら

ないため、ターゲットは、イニシエータが上記第1、第2の処理のいずれを行ったのかを知ることができない。従ってこのような場合にバスリセット発生時点の続きからデータ転送を開始してしまうと、誤ったデータ転送が行われてしまう可能性がある。

5 そこで本実施形態では図16に示すように、バスリセットの発生が要因となってイニシエータからアクノリッジメントが返って来なかつた場合には、デッド状態（データ転送不可状態）に遷移する。このようにすることで、誤ったデータ転送が行われる事態を防止できる。

さて、以上では、ターゲットがプリンタである場合について主に説明したが、
10 ターゲットがスキャナである場合には以下のような問題がある。

即ち、スキャナでは、一旦ヘッドが動き出すと、ヘッドを元に戻して再度同じデータを取得（スキャン）することはできないという問題がある。

そこで本実施形態では図17に示すように、スキャナエンジン（広義には上層のデバイス）からの転送データの中で、バスリセットの発生時点においてイニシエータに未だ転送していない転送データについては、破棄せずに保持するようになっている。即ち図17のH1に示すように、スキャナエンジンから取り込んでデータバッファに格納したが、IEEE1394のバスを介してイニシエータに未だ転送していないデータについては、データバッファをクリアしないようにして破棄せずに保持しておく。

20 このようにすれば、バスリセット発生後に図11に示す手法でバスリセット発生時点の続きからデータ転送を再開した場合にも、バスリセット発生前に取り込んだ画像データが失われてしまう等の不具合を防止できる。

5. 詳細な処理例

次に本実施形態の詳細な処理例について図18～図22のフローチャートを用いて説明する。

図18～図20は、バスリセット発生時（リコネクト時）の処理の詳細例を示すフローチャートである。

バスリセットが発生すると、ターゲットは、まず、イニシエータがログイン

しているか否かを判断し（ステップS20）、ログインしている場合にはIEEE1394のバス上の全ての転送処理（トランザクション）をキャンセルする（ステップS21）。一方、ログインしていない場合には、バスリセットが発生しても特別な処理は不要であるため、何もしない（ステップS22）。

5 次に、既にバスリセット処理に入っているか否かを判断する（ステップS23）。これにより、バスリセットが複数回発生した場合にそれに対応するバスリセット処理が無用に複数回繰り返される事態を防止できる。

10 次に、バスリセット発生時点のACK（アクノリッジメント）の状態を記憶しておく（ステップS24）。これにより、その後に発生するトランザクション（例えばリコネクトのトランザクション）により、バスリセット直後のACKの内容が消されてしまう事態を防止できる。

15 次に、IEEE1394のバス上で転送済みのデータのサイズ（バイト数）を記憶する（ステップS25）。即ち、バスリセット発生時点において処理中であったセグメントでの転送済みのデータのサイズ（図14のDS2）を記憶する。そして、ステップS23の判断のために、バスリセット処理中であることを示すフラグをオンにする（ステップS26）。即ち、このフラグがオンになると、その後に、バスリセットが発生しても、ステップS24～S26の処理は行われない。

20 次に、イニシエータからのリコネクト待ちになり（ステップS27）、イニシエータによりリコネクトされたか否かを判断する（ステップS28）。そして、リコネクトされなかった場合には、ログインORBのリコネクトフィールドで指定されたリコネクトタイムアウト時間が経過したか否かを判断する（ステップS29）。そして、経過した場合には、図13で説明した継続フラグ（データ転送が継続して再開される可能性があることを示すフラグ）をオフにして（ステップS30）、ログアウト状態に状態遷移する（ステップS31）。

25 一方、リコネクトタイムアウト時間内にリコネクトされた場合には、リコネクトしてきたイニシエータが、バスリセット前にログインしていたイニシエータか否かを判断し（ステップS32）、バスリセット前と異なるイニシエータ

であった場合には、そのイニシエータのリコネクトを拒否し、リコネクト待ちに戻る。

バスリセット前と同じイニシエータがログインしてきた場合には、印刷のためのコマンドブロックORB（印刷コマンドを含むORB）を、バスリセット発生時点において処理中だったか否かを判断する（図19のステップS33）。
5 そして、処理中でなかった場合には、継続フラグをオフにして（ステップS36）、アイドル状態に状態遷移する（ステップS37）。

一方、印刷のためのコマンドブロックORBを処理中であった場合には、ステータスのライト中（ステータスをライトしてからACKが返ってくるまでの期間）にバスリセットが発生したか否かを判断する（ステップS34）。そして、ステータスのライト中にバスリセットが発生した場合には、図18のステップS24で記憶したACKの情報に基づいて、ACKコンプリートか否かを判断する（ステップS35）。

そして、ACKコンプリートの場合には、継続フラグをオフにして（ステップS36）、アイドル状態に状態遷移する（ステップS37）。一方、ACKコンプリートでない場合には、ACKミッシングか否かを判断する（ステップS38）。そして、ACKミッシングでなければ何もせず（ステップS39）、ACKミッシングの場合には、継続フラグをオフにして（ステップS40）、図16で説明したようにデッド状態（データ転送不可状態）に状態遷移する（ステップS41）。

ステップS34でバスリセットの発生がステータスのライト中でないと判断された場合には、処理中のORBの転送（印刷）データを1バイトでも後段のプリンタエンジンに転送したか否かを判断する（ステップS42）。そして、1バイトも転送していなかった場合には継続フラグをオフにし（ステップS43）、アイドル状態に状態遷移する（ステップS49）。

一方、1バイトでもプリンタエンジンに転送していた場合には、ORBの内容（データサイズ、ページテーブル存在フラグP、コマンドブロック）や、バ
スリセット発生時点までに転送できたデータのサイズを記憶する（ステップS

44)。

このデータのサイズは、バスリセット発生時点で後段のプリントエンジンに既に転送したデータのバイト数と、バスリセット発生時点で IEEE1394 のバス上でのデータ転送は既に完了し、後段のプリントエンジンに転送中又はこれから転送する予定のデータのバイト数の合計に相当する。即ち、例えば、プリントで既に印刷したデータのバイト数と、プリントで現在印刷中又はこれから印刷予定のデータのバイト数の合計に相当する。

次に、ページテーブルが存在するか否かを判断し（ステップ S45）、存在しない場合には、ORBのデータスクリプタの内容を記憶する（ステップ S46）。即ち、ページテーブルが存在しない場合には、直接アドレス指定の場合の転送データのアドレス及びデータ長が記憶される（図 6C 参照）。

一方、ページテーブルが存在する場合には、図 14 で説明したように、ページテーブルの先頭セグメントの内容（アドレス、データ長）と、バスリセット発生時点で処理中だったセグメントの内容（アドレス、データ長）及びセグメント番号を記憶する（ステップ S47）。そして、図 13 で説明した継続フラグをオンにして（ステップ S48）、アイドル状態に状態遷移する（ステップ S49）。

このように本実施形態では、データ転送期間中にバスリセットが発生したと判断した場合に（図 19 のステップ S33、S34、図 20 のステップ S42）、継続フラグがオンになる（ステップ S48）。

図 21、図 22 は、通常時の処理の詳細例を示すフローチャートである。

まず、イニシエータからORBのリードを指示されたか否か（ドアベルレジスタがリングされたか否か）を判断し（ステップ S51）、指示されなかった場合にはアイドル状態にとどまる（ステップ S50）。一方、指示された場合には、イニシエータが作成したORBをイニシエータからリードする（ステップ S52）。そして、ORBが含むページテーブル存在フラグ P に基づいて、ページテーブルが存在するか否かを判断する（ステップ S53）。そして、ページテーブルが存在する場合には、ページテーブルのセグメントを例えば 8 ページテーブルが存在する場合には、ページテーブルのセグメントを例えれば 8 セ

グメントずつリードする（ステップS 5 4）。

次に、ORBのコマンドブロックにあるオペレーションコードに基づいて、図15で説明したように、リードしたORBが印刷のためのコマンドブロックORBか否かを判断する（ステップS 5 5）。そして、印刷のためのコマンドブロックORBであった場合には、ステップS 5 4でリードした8セグメントが、ページテーブルの最初の8セグメント（先頭セグメントを含む8セグメント）か否かを判断し（ステップS 5 6）、最初の8セグメントであった場合には、図22に示すコマンド・アドレス比較処理に移行する（ステップS 5 7）。

ステップS 5 5で印刷のためのコマンドブロックORBではないと判断された場合、ステップS 5 6で最初の8セグメントでないと判断された場合、及び、ステップS 5 7のコマンド・アドレス比較処理が終了した場合には、データのリード／ライトを行う（ステップS 5 8）。そして、1セグメント分のデータ、8セグメント分のデータをリード／ライトするまで繰り返す（ステップS 5 9、S 6 0）。

次に、ページテーブルの全てのセグメントをリード／ライトしたか否かを判断し（ステップS 6 1）、全てのセグメントをリード／ライトしていない場合には、ページテーブルの次の8セグメントをリードする（ステップS 5 4）。一方、ページテーブルの全てのセグメントをリードした場合には、イニシエータに対してステータスをライトする（ステップS 6 2）。そして、印刷物の印刷のための全てのORBをリードしたか否かを判断し（ステップS 6 3）、次のORBがある場合にはステップS 5 2に戻り、次のORBが無い場合にはアイドル状態に状態遷移する（ステップS 5 0）。

ステップS 5 3でページテーブルが存在しないと判断した場合には、リードしたORBが印刷のためのコマンドブロックORBか否かを判断する（ステップS 6 4）。そして、印刷のためのコマンドブロックORBであった場合には、図22に示すコマンド・アドレス比較処理に移行する（ステップS 6 5）。

一方、印刷のためのコマンドブロックORBではないと判断された場合、及び、コマンド・アドレス比較処理が終了した場合には、データをリード／ライ

トし(ステップS 6 6)、全てのデータをリード/ライトするまで繰り返す(ステップS 6 7)。そして、全てのデータをリード/ライトした場合には、ステップS 6 2に移行し、イニシエータに対してステータスをライトする。

図22のコマンド・アドレス比較処理においては、まず、継続フラグがオンか否かを判断する(ステップS 7 0)。この継続フラグは、図20のステップS 4 8においてオンにされるフラグである。そして、継続フラグがオフの場合にはステップS 7 6に移行して、転送データの先頭アドレス(ページテーブルの先頭セグメントのアドレス)を記憶し、コマンド・アドレス比較処理を終了する。

継続フラグがオンの場合には、図11、図12で説明したように、リードしたORBの内容がバスリセット前のORBの内容と同一か否かを判断する(ステップS 7 1)。この場合に、比較対象となるバスリセット前のORBの内容は、図20のステップS 4 4において記憶されている。また、本実施形態では、アドレス比較(ステップS 7 2)に先立って、ORBの内容比較(ステップS 7 1)を行っている。

ORBの内容がバスリセット前と同一であった場合には、転送データの先頭アドレスがバスリセット前と同一か否かを判断する(ステップS 7 2)。そして、同一の場合には、データ転送の設定をバスリセット発生前の状態に戻す(ステップS 7 3)。即ち、図20のステップS 4 4で記憶したバスリセット発生時点までの転送済みデータサイズや、ステップS 4 7で記憶したセグメントの内容やセグメント番号などに基づいて、図11のD 5に示す位置(バスリセット発生時点の続き)からデータ転送を再開できるように、データ転送の設定をバスリセット前の状態に戻す。そして、継続フラグをオフに戻す(ステップS 7 4)。この場合に、図17で説明したように、バスリセット前に既に転送を完了していたデータが消失しないように、ターゲットのデータバッファ上のデータをクリアしないようにする。

なお、ステップS 7 4の後に、ステップS 7 6のように転送データの先頭アドレスを記憶しておかなければ、バスリセット発生時点の続きからデータ転送

を再開する場合には、バスリセット発生前に記憶しておいた先頭アドレスがそのまま使えるからである。

ステップS71でORBの内容がバスリセット前と同一でないと判断した場合、或いはステップS72で先頭アドレスがバスリセット前と同一でないと判断した場合には、データ転送の再開処理を行わず、継続フラグをオフに戻すと共に転送データの先頭アドレスを記憶しておく（ステップS75、S76）、即ち、この場合には、リードしたORBを全くの新規のORBとして処理することになる。

なお、ステップS75の場合には、リードしたORBを最初から処理することになるため、ステップS74と異なり、ターゲットのデータバッファのデータをクリアする。

6. 電子機器

次に、本実施形態のデータ転送制御装置を含む電子機器の例について説明する。

例えば図23Aに電子機器の1つであるプリンタの内部ブロック図を示し、図24Aにその外観図を示す。CPU（マイクロコンピュータ）510はシステム全体の制御などを行う。操作部511はプリンタをユーザが操作するためのものである。ROM516には、制御プログラム、フォントなどが格納され、RAM518はCPU510のワーク領域として機能する。表示パネル519はプリンタの動作状態をユーザに知らせるためのものである。

PHYデバイス502、データ転送制御装置500を介して、パーソナルコンピュータなどの相手ノードから送られてきた印字データは、バス504を介して印字処理部（プリンタエンジン）512に直接送られる。そして、印字データは、印字処理部512にて所与の処理が施され、プリントヘッダなどからなる印字部（データを出力するための装置）514により紙に印字されて出力される。

図23Bに電子機器の1つであるスキャナの内部ブロック図を示し、図24Bにその外観図を示す。CPU520はシステム全体の制御などを行う。操作

部 521 はスキャナをユーザが操作するためのものである。ROM 526 には制御プログラムなどが格納され、RAM 528 は CPU 520 のワーク領域として機能する。

光源、光電変換器などからなる画像読み取り部（データを取り込むための装置）522 により原稿の画像が読み取られ、読み取られた画像のデータは画像処理部（スキャナエンジン）524 により処理される。そして、処理後の画像データがバス 505 を介してデータ転送制御装置 500 に直接送られる。データ転送制御装置 500 は、この画像データにヘッダなどを付加することでパケットを生成し、PHY デバイス 502 を介してパーソナルコンピュータなどの相手ノードに送信する。

図 23C に電子機器の 1 つである CD-RW ドライブの内部ブロック図を示し、図 24C にその外観図を示す。CPU 530 はシステム全体の制御などを行う。操作部 531 は CD-RW をユーザが操作するためのものである。ROM 536 には制御プログラムなどが格納され、RAM 538 は CPU 530 のワーク領域として機能する。

レーザ、モータ、光学系などからなる読み取り & 書き込み部（データを取り込むための装置又はデータを記憶するための装置）533 により CD-RW 532 から読み取られたデータは、信号処理部 534 に入力され、エラー訂正処理などの所与の信号処理が施される。そして、信号処理が施されたデータが、バス 506 を介してデータ転送制御装置 500 に直接送られる。データ転送制御装置 500 は、このデータにヘッダなどを付加することでパケットを生成し、PHY デバイス 502 を介してパーソナルコンピュータなどの相手ノードに送信する。

一方、PHY デバイス 502、データ転送制御装置 500 を介して、相手ノードから送られてきたデータは、バス 506 を介して信号処理部 534 に直接送られる。そして、信号処理部 534 によりこのデータに所与の信号処理が施され、読み取り & 書き込み部 533 により CD-RW 532 に記憶される。

なお、図 23A、図 23B、図 23C において、CPU 510、520、5

30 の他に、データ転送制御装置 500 でのデータ転送制御のための C P U を別に設けるようにしてもよい。

また、図 23 A、図 23 B、図 23 C では R A M 501 (データバッファに相当) がデータ転送制御装置 500 の外部に設けられているが、R A M 501 をデータ転送制御装置 500 に内蔵させてもよい。

本実施形態のデータ転送制御装置を電子機器に用いれば、バスに新たな電子機器が接続されてバスリセットが発生した場合にも、バスリセットを原因とする不具合の発生が防止される。これにより、電子機器の誤動作を防止できる。

また本実施形態のデータ転送制御装置を電子機器に用いれば、高速なデータ転送が可能になる。従って、ユーザがパーソナルコンピュータなどによりプリントアウトの指示を行った場合に、少ないタイムラグで印字が完了するようになる。また、スキャナへの画像取り込みの指示の後に、少ないタイムラグで読み取り画像をユーザは見ることができるようになる。また、C D - R Wからのデータの読み取りや、C D - R Wへのデータの書き込みを高速に行うことができるようになる。

また本実施形態のデータ転送制御装置を電子機器に用いることで、C P U 上で動作するファームウェアの処理負担が軽減され、安価な C P U や低速のバスを用いることが可能になる。更に、データ転送制御装置の低コスト化、小規模化を図れるため、電子機器の低コスト化、小規模化も図れるようになる。

なお本実施形態のデータ転送制御装置を適用できる電子機器としては、上記以外にも例えば、種々の光ディスクドライブ (C D - R O M、D V D)、光磁気ディスクドライブ (M O)、ハードディスクドライブ、T V、V T R、ビデオカメラ、オーディオ機器、電話機、プロジェクタ、パーソナルコンピュータ、電子手帳、ワードプロセッサなど種々のものを考えることができる。

なお、本発明は本実施形態に限定されず、本発明の要旨の範囲内で種々の変形実施が可能である。

例えば、本発明のデータ転送制御装置の構成は、図 7 に示す構成が特に望ましいが、これに限定されるものではない。

また、データ転送期間中にバスリセットが発生したか否かの判断手法、コマンドの比較手法、データ転送の再開の手法も、本実施形態で説明した手法が特に望ましいが、これに限定されるものではない。

また、本発明は IEEE 1394におけるバスリセットに特に有用だが、これ以外にも、少なくともノードのトポロジー情報をクリアするようなりセットであれば適用できる。

また、本発明は、IEEE 1394規格でのデータ転送に適用されることが特に望ましいが、これに限定されるものではない。例えば IEEE 1394と同様の思想に基づく規格や IEEE 1394を発展させた規格におけるデータ転送にも本発明は適用できる。

クレーム

1. バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、

5 相手ノードとの間でデータ転送が開始してから完了するまでの間のデータ転送期間中に、ノードのトポロジ情報をクリアするリセットが発生したか否かを判断する判断手段と、

10 該リセットの発生前に相手ノードから転送されてきたデータ転送オペレーション要求のための第1のコマンドパケットの内容と、該リセットの発生後に相手ノードから転送されてきたデータ転送オペレーション要求のための第2のコマンドパケットの内容とを比較するコマンド比較手段と、

15 ノードのトポロジ情報をクリアするリセットが前記データ転送期間中に発生したと判断され、且つ、前記第1、第2のコマンドパケットが同一内容であると判断された場合には、リセット発生時点のデータ転送の続きをデータ転送を再開する再開手段と、

20 15 を含むことを特徴とするデータ転送制御装置。

2. 請求項1において、

前記判断手段が、

25 ノードのトポロジ情報をクリアするリセットの発生時点においてデータ転送オペレーション要求のための前記第1のコマンドパケットを処理中であり、且つ、該第1のコマンドパケットによるデータ転送が既に行われておりデータ転送完了のステータスを相手ノードに転送していない場合に、該リセットが前記データ転送期間中に発生したと判断することを特徴とするデータ転送制御装置。

3. 請求項1において、

前記判断手段が、

25 ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生したと判断した場合に、データ転送が継続して再開される可能性があることを示す継続フラグをオンにセットすることを特徴とするデータ転送制御装置。

4. 請求項1において、

データ転送を再開するアドレスを特定するための情報と、データ転送オペレーション要求のための前記第1のコマンドパケットの内容とを、該リセットの発生後、データ転送が再開されるまでの間に記憶するコマンド記憶手段を含むことを特徴とするデータ転送制御装置。

5. 請求項1において、

前記コマンド比較手段が、

ノードのトポロジ情報をクリアするリセットの発生後に相手ノードから転送されてきたコマンドパケットの中で最初に転送されてきたデータ転送オペレーション要求のためのコマンドパケットを、前記第1のコマンドパケットの比較対象となる前記第2のコマンドパケットとして採用することを特徴とするデータ転送制御装置。

6. 請求項1において、

データ転送完了のステータスを相手ノードに転送したが、ノードのトポロジ情報をクリアするリセットの発生が要因となって相手ノードからアクノリッジメントが返って来なかった場合に、データ転送不可状態に状態遷移することを特徴とするデータ転送制御装置。

7. 請求項1において、

上層のデバイスから転送されてきた転送データの中で、ノードのトポロジ情報をクリアするリセットの発生時点において相手ノードへ未だ転送していない転送データについては、破棄せずに保持しておくことを特徴とするデータ転送制御装置。

8. 請求項1において、

前記リセットが、IEEE1394の規格において定義されるバスリセットであることを特徴とするデータ転送制御装置。

25 9. 請求項1のデータ転送制御装置との間でのデータ転送を制御するためのプログラムを含む、コンピュータ使用可能な情報記憶媒体であって、

ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した場合には、該リセットの発生前に転送されたデータ転送オペレーション要求の

ための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする情報記憶媒体。

10. 請求項2のデータ転送制御装置との間でのデータ転送を制御するための
5 プログラムを含む、コンピュータ使用可能な情報記憶媒体であって、
ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した
場合には、該リセットの発生前に転送されたデータ転送オペレーション要求の
ための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、
データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする
10 情報記憶媒体。

11. 請求項3のデータ転送制御装置との間でのデータ転送を制御するための
プログラムを含む、コンピュータ使用可能な情報記憶媒体であって、
ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した
場合には、該リセットの発生前に転送されたデータ転送オペレーション要求の
15 ための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、
データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする
情報記憶媒体。

12. 請求項4のデータ転送制御装置との間でのデータ転送を制御するための
プログラムを含む、コンピュータ使用可能な情報記憶媒体であって、
ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した
場合には、該リセットの発生前に転送されたデータ転送オペレーション要求の
ための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、
データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする
20 情報記憶媒体。

25 13. 請求項5のデータ転送制御装置との間でのデータ転送を制御するための
プログラムを含む、コンピュータ使用可能な情報記憶媒体であって、
ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した
場合には、該リセットの発生前に転送されたデータ転送オペレーション要求の

ための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする情報記憶媒体。

14. 請求項6のデータ転送制御装置との間でのデータ転送を制御するための
5 プログラムを含む、コンピュータ使用可能な情報記憶媒体であって、

ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した場合には、該リセットの発生前に転送されたデータ転送オペレーション要求のための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする
10 情報記憶媒体。

15. 請求項7のデータ転送制御装置との間でのデータ転送を制御するための
プログラムを含む、コンピュータ使用可能な情報記憶媒体であって、

ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した場合には、該リセットの発生前に転送されたデータ転送オペレーション要求のための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする
15 情報記憶媒体。

16. 請求項8のデータ転送制御装置との間でのデータ転送を制御するための
プログラムを含む、コンピュータ使用可能な情報記憶媒体であって、

ノードのトポロジ情報をクリアするリセットがデータ転送期間中に発生した場合には、該リセットの発生前に転送されたデータ転送オペレーション要求のための第1のコマンドパケットと同一内容の第2のコマンドパケットを作成し、データ転送制御装置に転送要求するためのプログラムを含むことを特徴とする
20 情報記憶媒体。

25 17. 請求項1のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに所与の処理を施す装置と、

処理が施されたデータを出力又は記憶するための装置とを含むことを特徴と

する電子機器。

18. 請求項 2 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに
所与の処理を施す装置と、

5 处理が施されたデータを出力又は記憶するための装置とを含むことを特徴と
する電子機器。

19. 請求項 3 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに
所与の処理を施す装置と、

10 处理が施されたデータを出力又は記憶するための装置とを含むことを特徴と
する電子機器。

20. 請求項 4 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに
所与の処理を施す装置と、

15 处理が施されたデータを出力又は記憶するための装置とを含むことを特徴と
する電子機器。

21. 請求項 5 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに
所与の処理を施す装置と、

20 处理が施されたデータを出力又は記憶するための装置とを含むことを特徴と
する電子機器。

22. 請求項 6 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに
所与の処理を施す装置と、

25 处理が施されたデータを出力又は記憶するための装置とを含むことを特徴と
する電子機器。

23. 請求項 7 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードから受信したデータに

所与の処理を施す装置と、

処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とする電子機器。

24. 請求項8のデータ転送制御装置と、

5 前記データ転送制御装置及びバスを介して相手ノードから受信したデータに所与の処理を施す装置と、

処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とする電子機器。

25. 請求項1のデータ転送制御装置と、

10 前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子機器。

26. 請求項2のデータ転送制御装置と、

15 前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子機器。

27. 請求項3のデータ転送制御装置と、

20 前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子機器。

28. 請求項4のデータ転送制御装置と、

25 前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子機器。

29. 請求項 5 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所
与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子
機器。

30. 請求項 6 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所
与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子
機器。

31. 請求項 7 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所
与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子
機器。

32. 請求項 8 のデータ転送制御装置と、

前記データ転送制御装置及びバスを介して相手ノードに転送するデータに所
与の処理を施す装置と、

処理が施されるデータを取り込むための装置とを含むことを特徴とする電子
機器。

開示内容の要約

ノードのトポロジ情報をクリアするリセットが発生した場合に生じる不具合を解消できるデータ転送制御装置、情報記憶媒体及び電子機器を提供することが目的である。I E E E 1 3 9 4 規格のデータ転送制御装置において、データ転送期間中にバスリセットが発生し且つバスリセットの前後のO R Bの内容が同一の場合に、バスリセット発生時点の続きからデータ転送を再開し、プリンタの二重印刷を防止する。バスリセットがデータ転送期間中に発生した場合に継続フラグをオンにする。バスリセット後に最初に転送されてきた、印刷コマンドを含むコマンドブロックO R Bを、バスリセット前のO R Bの比較対象として採用する。バスリセットが要因となりイニシエータからA C Kが返送されない場合に、デッド状態に状態遷移する。スキヤナからの転送データの中で、バスリセット発生時点においてイニシエータに未だ転送していない転送データについては、破棄せずに保持しておく。